

Nonstationary Predictive Deconvolution

Gary F. Margrave and Michael P. Lamoureux

ABSTRACT

We present a general method for construction of a uniform partition of unity (POU) that is exact over a finite segment of the real line and whose individual windows are constructed from asymmetric Gaussians. The spacing between windows need have no relationship to the Gaussian half-width. We illustrate the uses of such POU's to decompose a signal into a discrete set of temporally localized signals which we call Gabor slices. We then apply this theory to construct a time-domain nonstationary deconvolution method based on gapped prediction filtering. We call this new method *slicedecon* because it operates directly on the individual Gabor slices. We also prescribe the construction of nonstationary autocorrelation functions as an analysis tool. We then compare *slicedecon* with the more established Gabor deconvolution or *gabordecon*. When the prediction filtering is unit-lag, we show that *slicedecon* achieves results comparable to *gabordecon* on a nonstationary (Q attenuation) synthetic. For lags greater than unity *slicedecon* appears to suppress, though not eliminate, periodicities in the nonstationary autocorrelation of a signal. Testing on a synthetic with multiples has not yet indicated any dramatic elimination of the unwanted multiple reflections.

INTRODUCTION

Predictive deconvolution (Peacock and Treitel, 1969) has long been used, with limited success, as a method of multiple suppression. The original theory uses stationary prediction filters to estimate the predictable part of a time series which is then subtracted from the original signal to give the prediction error. When the prediction distance (or lag) is one sample, the prediction error is an estimate of the reflectivity. For greater prediction lags, predictive deconvolution has been shown to remove multiples under ideal circumstances. However in general the multiple content of a seismogram is nonstationary. Two major effects are at play here. First, even simple multiples, like those from a hard water bottom, are easily shown to be periodically spaced in time only on a zero offset trace, and only if the water bottom is flat. Second, a given interface generates multiples that arrive later in time than the primary. This means that the multiple train trailing behind the primary seismic pulse grows as the pulse progresses. Taner (1980) proposed predictive deconvolution in the tau-p domain as a remedy for the first effect. Later other similar ideas have been tested such as predictive deconvolution in the radial trace domain (Perez and Henley, 2000).

Recently, we have developed a nonstationary spiking deconvolution in the Gabor domain (e.g. Margrave and Lamoureux, 2001, Margrave et al, 2004) which has proven very successful in dealing with the nonstationary effects of anelastic attenuation. Gabor deconvolution is a natural extension of Wiener spiking deconvolution (Robinson and Treitel, 1967) to the Gabor time-frequency domain. Here we report on our initial attempts to do something similar with the closely related predictive deconvolution. This development has been hindered due to the lack of a useful theory of gapped predictive deconvolution in the frequency domain. Here we sidestep this issue by developing the

theory in the nonstationary “time-time” domain. This is an intermediate domain realized in the process of a Gabor transform after localizing a signal with a temporal window and before Fourier transformation. In our implementation, we use a partition of unity (POU) to decompose a signal into “Gabor slices” which are localized time signals that sum to recreate the original signal. Then on each Gabor slice we implement a conventional stationary prediction operator. This is not as desirable as having a true nonstationary prediction filter but it is a first approximation to one, and our approach as the correct stationary limit.

Our paper begins with a development of POU theory that is a precise decomposition on a finite segment of the real line. We use uniform windows of constant spacing where the basic window is an asymmetric Gaussian. We show how these windows can be normalized to make the POU exact. Then we develop the Gabor slice decomposition of a signal and the implementation of a nonstationary operator on the slices. In particular, if the operator on each slice is a prediction filter, then we have a predictive deconvolution. We also show how to construct a nonstationary autocorrelation. Then we test our new algorithm on a nonstationary primaries only synthetic with Q effects and on a lossless synthetic with nonstationary multiple effects. As a comparison we also show Gabor deconvolution of these synthetic data. When the prediction distance is a single sample, our new method performs almost as well as Gabor decon. For more general lags, it appears able to suppress periodicities in the autocorrelation but we see only minimal reduction in multiple content.

THEORY

Partitions of Unity

Consider a subset of the real line $\sigma = [a, b], b > a$ and a finite set of functions, $\Omega_k(x), k \in K = [1, N]$ with the property

$$\sum_{k=1}^N \Omega_k(x) = 1, \forall x \in \sigma. \quad (1)$$

The set $\{\Omega_k\}$ is called a partition of unity or POU. Sometimes further constraints are placed on the $\{\Omega_k\}$ such that they be smooth (differentiable up to some order) or that their support be confined to a subset of σ . In the present context, we do not need these constraints but we do require that each Ω_k be nonnegative everywhere, that is,

$$\Omega_k(x) \geq 0 \quad \forall x \in \sigma, k \in K. \quad (2)$$

Given such a POU, we define the *analysis window*, $g_k(x)$, and the *synthesis window*, $\gamma_k(x)$, through

$$g_k(x) = \Omega_k^p(x), \quad p \in [0,1] \quad (3)$$

and

$$\gamma_k(x) = \Omega_k^{1-p}(x), \quad p \in [0,1]. \quad (4)$$

Since $g_k \gamma_k = \Omega_k$, the analysis and synthesis windows are a useful factorization of a POU that will be exploited in the next section of this paper.

The construction of a POU can be done in many ways. In the simplest schemes, the Ω_k are all identical in shape and are simply translated (shifted) versions of an elemental window. This is called a uniform partition. In more sophisticated applications, extremely complicated windows can be constructed, each one unique from the others, yet they still sum precisely to unity. Such partitions are called non-uniform or sometimes adaptive (see Ma and Margrave 2008 or Wards et al 2008 for examples). As a nearly trivial example of a uniform POU, let $I_{c,d}(x)$ be the indicator function, or boxcar function, defined by

$$I_{c,d}(x) = \begin{cases} 1, & x \in [c, d), d > c \\ 0, & \text{otherwise} \end{cases}. \quad (5)$$

Then

$$\Omega_k(x) = I_{a+(k-1)\delta, a+k\delta}(x) \quad (6)$$

where $\delta = (b-a)/N$ defines a POU on σ consisting of N boxcars.

More commonly, a POU can be built by a set of translated Gaussians (Margrave and Lamoureux, 2002). That is let,

$$\bar{\Omega}_k(x) = C \exp\left(-\frac{(x - k\delta)^2}{\lambda^2}\right) \quad (7)$$

where C is a normalization constant to be determined and the reason for the overbar will soon be apparent. The set $\{\bar{\Omega}_k\}$ forms an approximate POU provided that $\delta \ll \lambda$ (Margrave and Lamoureux, 2002). However, we can do better than the approximate POU of equation (7) if we relax the requirement that each window be a simple translate of a fundamental window. We simply define

$$\Omega_k(x) = \frac{\bar{\Omega}_k(x)}{\Theta(x)} \tag{8}$$

where

$$\Theta(x) = \sum_{j=1}^N \bar{\Omega}_j(x) \tag{9}$$

is called the *normalization factor*, and the partition formed by $\{\Omega_k\}$ is now exact. Furthermore, the constant C of equation (7) is now irrelevant. Formed in this way, $\{\Omega_k\}$ is not strictly a uniform POU but it is based on an underlying uniform window set. Calling this an adaptive POU might be misleading since there is no adaptation to a physical parameter, rather, the windows are non-uniform purely to make the POU exact over the finite interval σ .

Figure 1 shows the construction of a POU on $\sigma = [0,1]$ using underlying equally spaced Gaussians. In this case, the spacing of the Gaussians is small compared to their width (i.e. $\delta \ll \lambda$) and the normalization factor, $\Theta(x)$, is mostly constant except near the ends of $\sigma = [0,1]$. Thus the main effect of equation (8) is to alter the windows near the boundary. Figure 2 shows a different situation where $\delta = 2\lambda$. Now the normalization factor shows both end effects and ripples throughout. The normalized windows produced by equation (8) are wider, with steeper slopes, than the original Gaussians. As the spacing between Gaussians becomes much larger than their width, the normalized windows begin to look more like rounded boxcars than Gaussians.

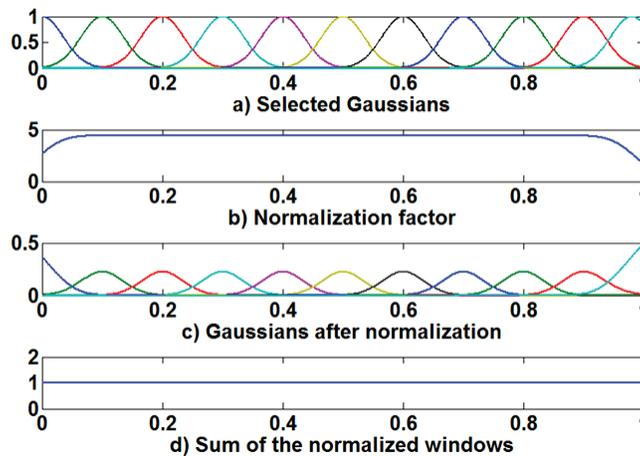


FIG. 1. Construction of a uniform POU. a) Every 5th window of a set of uniform Gaussians forming $\{\bar{\Omega}_k\}$. b) The normalization factor as expressed by equation (9). c) Every 5th window of the partition $\{\Omega_k\}$ as expressed by equation (8). d) The sum of $\{\Omega_k\}$ showing perfect summation to unity.

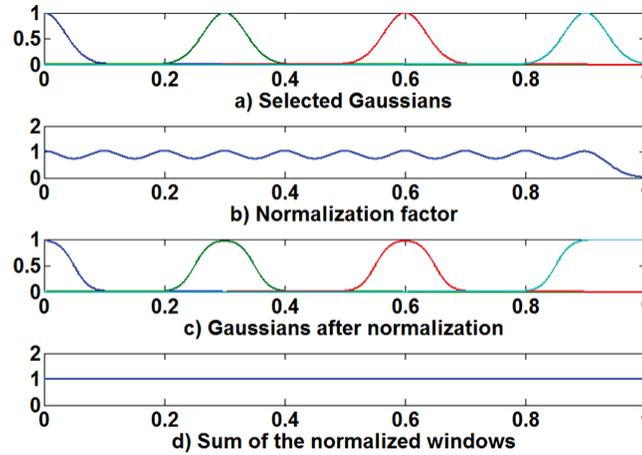


FIG. 2. Similar to Figure 1 except that the Gaussians are spaced more coarsely and every 3rd window is plotted.

Any set of non-negative bump functions whose normalization factor (equation (9)) never vanishes can be used to construct an exact POU. Of interest in deconvolution are asymmetric windows that are wider at times greater than the window center time than at times less than the center time. For example, consider the asymmetric Gaussian defined by

$$\bar{\Omega}_k(x) = \begin{cases} C \exp\left(-\frac{(x - k\delta)^2}{\lambda_1^2}\right), & x \leq k\delta \\ C \exp\left(-\frac{(x - k\delta)^2}{\lambda_2^2}\right), & x > k\delta \end{cases} \quad (10)$$

Figure 3 shows a POU constructed from such functions with $\lambda_1 = 0.02$, $\lambda_2 = 0.05$, $\delta = 0.02$. Again the POU can be made exact and the asymmetric Gaussian is only distorted significantly near the ends of the interval.

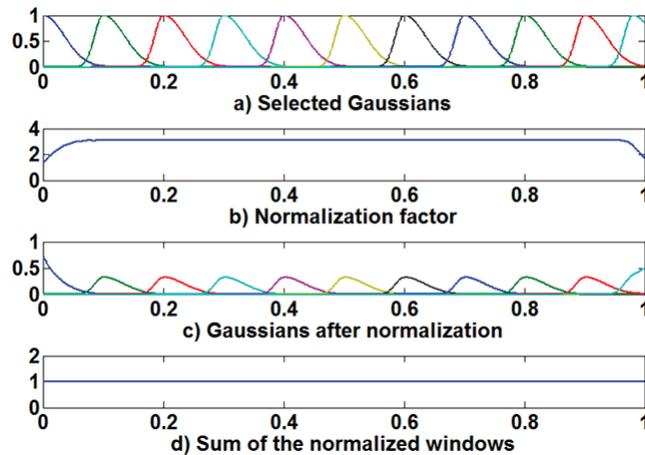


FIG. 3. Similar to Figures 1 and 2 except that the asymmetric Gaussians of equation (10) are used. Every 5th window is shown.

The Gabor slice, signal reconstruction, and nonstationary filters

Using the POU concepts of the previous section, we can decomposed any signal (here this means any 1D function of bounded energy, i.e. in L^2) into a suite of *Gabor slices* defined by

$$s_k(t) = g_k(t)s(t), \quad (11)$$

and the signal can be reconstructed from its slices, $\{s_k(t)\}$, by

$$s(t) = \sum_{k=1}^N \gamma_k(t) s_k(t). \quad (12)$$

In these equations, $g_k \gamma_k = \Omega_k$ and $\sum_k \Omega_k = 1$. The proof of equation (12) is an immediate consequence of these properties.

The decomposition of a signal into Gabor slices provides a very flexible mechanism for nonstationary, or time-variant, signal analysis and processing. Let $T_k : L^2 \rightarrow L^2$ be any linear operator that maintains the finite energy of a signal, and the subscript k indicates that the operator can depend on window position. Then we define the nonstationary operator

$$(\mathbf{T}s)(t) = \sum_{k=1}^N \gamma_k T_k s_k = \sum_{k=1}^N \gamma_k T_k g_k s. \quad (13)$$

For example, if $T_k = \mathbf{F}^{-1} \mathbf{F}$, $\forall k$ (where \mathbf{F} is the Fourier transform from time, t , to frequency, ω) then $\mathbf{T}s = s$ is the identity operation but expressed in a very useful form as

$$s = (\mathbf{T}s)(t) = \sum_{k=1}^N \gamma_k T_k s_k = \sum_{k=1}^N \gamma_k \mathbf{F}^{-1} \mathbf{F} g_k s. \quad (14)$$

Equation (14) actually expresses decomposition of the signal by the forward Gabor transform defined as (we drop the explicit detail of the Fourier variables)

$$\mathbf{G}_k s(t) = \mathbf{F} g_k s(t) = \hat{s}_k(t) \quad (15)$$

and the subsequent reconstruction of the signal by the inverse Gabor transform as

$$s = \mathbf{G}_k^{-1} \mathbf{G}_k s = \sum_{k=1}^N \gamma_k \mathbf{F}^{-1} \hat{s}_k(t) = \sum_{k=1}^N \gamma_k \mathbf{F}^{-1} \mathbf{F} g_k s(t). \quad (16)$$

At this point, our Gabor transform accomplishes nothing but if we now let $T_k = \mathbf{F}^{-1} M_{\hat{\alpha}_k} \mathbf{F}$, where M_{α_k} means multiplication by $\hat{\alpha}_k(\omega)$, then

$$(\mathbf{T}s)(t) = \sum_{k=1}^N \gamma_k \mathbf{F}^{-1} M_{\hat{\alpha}_k} \mathbf{F} s_k \quad (17)$$

expresses the processing of the signal in the Gabor domain by multiplication with the 2D function $\hat{\alpha}_k(\omega)$. Using the convolution theorem, we can re-express equation (17) as

$$(\mathbf{T}s)(t) = \sum_{k=1}^N \gamma_k \mathbf{C}_{\alpha_k} s_k = \sum_{k=1}^N \gamma_k \mathbf{C}_{\alpha_k} \mathbf{g}_k s \quad (18)$$

where \mathbf{C}_{α_k} means convolution (over time) with $\alpha_k(t)$. Both equations (17) and (18) are prescriptions for a nonstationary filter and are equivalent. However, the former is more useful for Gabor deconvolution (Margrave et al. 2004, Margrave and Lamoureux 2002) because the deconvolution operator, $\hat{\alpha}_k(\omega)$, is best designed in the time frequency domain. On the other hand, equation (18) is most useful in the present context because gapped prediction filtering is most easily done in the time domain.

Nonstationary deconvolution and prediction filtering

The word *deconvolution* is used in signal processing to refer to the process of convolving a time series with a filter that is the inverse of another filter, or wavelet in geophysics, that was previously convolved. For example, let $s = w \bullet r$ where all three symbols are signals and the bullet indicates convolution in time. Now suppose we can find another signal, d , with the property $d \bullet w = \delta$ where here δ is the Dirac Delta distribution which is the identity under convolution. Then, r can be recovered from s by convolution with d : $d \bullet s = d \bullet w \bullet r = \delta \bullet r = r$. Here we have a convolution that undoes a previous convolution and this is called *deconvolution*. The signal d is said to be the (convolutional) inverse of w .

Mathematically, it turns out that if w is a signal (i.e. has finite energy) then d cannot be. This might imply difficulty in constructing d but this is easily dealt with by relaxing the condition $d \bullet w = \delta$ to $d \bullet w = \tilde{\delta}$, where by $\tilde{\delta}$ we mean a so-called bandlimited Delta function. This means that $\tilde{\delta}$ is a signal whose Fourier amplitude spectrum is unity out to some finite frequency and then decays smoothly to zero, and whose Fourier phase spectrum is everywhere zero. The design of such bandlimited deconvolution *operators* or digital (sampled) signals was first solved by Norbert Wiener using least-squares methodology. Now a standard part of digital signal theory, Wiener's method allows the design of approximate deconvolution operators of any length that are optimal in a least-squares sense.

In exploration seismology, deconvolution is a standard tool used to remove the source signature, or wavelet, from seismograms (seismic signals). Here, two more problems arise. First, w is not known a priori and must be estimated from the signal itself; and, second, w evolves with time due to anelastic attenuation so the convolution $s = w \bullet r$ must be nonstationary. The estimation of w from s is usually accomplished by assuming that r is statistically random (it's autocorrelation is a Delta distribution) and by

assuming that w is minimum phase. A result from Wiener's theory is that the inverse of a minimum phase wavelet is completely specified by the autocorrelation of the wavelet. The autocorrelation of the wavelet can be estimated from the autocorrelation of s under the randomness assumption for r . These assumptions, while usually not strictly valid, yield estimates for d that have proven extremely useful to enhance seismograms.

The nonstationary nature of the seismogram is a more difficult problem. A common strategy is to use a *localization* of s (i.e. a time-limited segment of s) to estimate a d_{loc} and then form $d_{loc} \bullet s$ while simply accepting that the result is not optimal far from the localization zone. We have also had success with Gabor deconvolution where a complete set of localizations, that is a suite of Gabor slices, are used to build a Gabor multiplier that accomplishes a nonstationary deconvolution. In this paper, we will do something similar in that we will form a suite of Gabor slices, and on each slice we will design a prediction filter with certain properties. This filter defines the *predictable part* of the slice and the remainder is the unpredictable part. The synthesis of the unpredictable parts of all Gabor slices forms a nonstationary predictive deconvolution.

At this point, it will help to recall the definition of a prediction filter. Since this theory is usually formulated for sampled signals, we move our notation to discrete by letting continuous time t transition to discrete time denoted by the integer variable j . More explicitly $j \in [1, J]$ is the sample index, Δt is the sample size in physical units, and $t = (j-1)\Delta t$ defines the mapping from samples to continuous time. Then $s(t)$ will denote a continuous signal and $s(j)$ the corresponding sampled signal. Unless otherwise stated, i, j, k are all integer variable. A prediction filter is a K length, causal, digital signal, $u(k): k \in [1, K]$, with the property that

$$\sum_{j=1}^J (s(j+m) - (u \bullet s)(j))^2 = \min \quad (19)$$

where the integer m is called the prediction lag or distance. For $m = 1$, the prediction filter takes any K samples of s and tries to predict sample $K+1$. More generally, the prediction filter u is required to be causal meaning that it takes M points from s and combines them linearly to predict the value of s but m points in the future (greater times). The filter is also stationary and so prescribes the same linear weights for all times. Once a prediction filter has been designed, the predictable part of s is then defined as $u \bullet s$ and the unpredictable part, or prediction error, is found by subtracting the predicted part from the original signal. Let δ^m be a digital signal which is all zeros except for the single sample in the m^{th} position which is unity (e.g. δ^1 is a discrete delta function at the origin). Then we can write the unpredictable part of s as

$$s - \delta^m \bullet u \bullet s = (\delta^1 - \delta^m \bullet u) \bullet s. \quad (20)$$

The filter $(\delta^1 - \delta^m \bullet u)$ is called the *prediction error filter of lag m* for obvious reasons. It is known that the most common seismic deconvolution method, Wiener spiking deconvolution, is equivalent to prediction error filtering of lag 1 (Reference). Thus, prediction error filtering generalizes deconvolution to any prediction lag and such methods are generally called predictive deconvolution. Thus we define the predictive deconvolution operator of lag m as

$$d^m = \delta^1 - \delta^m \bullet u \quad (21)$$

where u has been designed as the least squares solution to equation (19). There is a long history of seismic data processing that attempts to use predictive deconvolution to attack various classes of multiple reflections. While successful and popular in many cases, the method often fails because the multiples are a nonstationary effect. Here we report on a nonstationary generalization of prediction error filtering.

Using the Gabor slicing theory developed in the previous section, we define the nonstationary predictive deconvolution of signal s by

$$s_d = \sum_{k=1}^N \gamma_k d_k^{m_k} \bullet s_k = \sum_{k=1}^N \gamma_k d_k^{m_k} \bullet (g_k s) \quad (22)$$

Where the notation $d_k^{m_k}$ emphasizes that both the design of the prediction filter and the value of the lag are unique for each Gabor slice.

The solution of equation (19) using linear algebra requires construction of the normal equations and the consequent construction of the autocorrelation of the signal. Essentially the autocorrelation determines the prediction operator and common interpretation procedure involves analyzing the autocorrelation for undesired periodicity (interpreted to be multiples) and choosing the prediction lag m and operator length K in the process. To replicate this process with our nonstationary method, we will need a time-variant autocorrelation defined by

$$a_k(\tau) = (\bar{s}_k \bullet s_k)(\tau) = (\overline{g_k s}) \bullet (g_k s)(\tau) \quad (23)$$

where the backwards arrow over a signal is used to denote time reversal, that is $\bar{s}(t) = s(-t)$. The time-variant autocorrelation of a 1D signal is a 2D function of window center time (k) and lag time (τ).

We have constructed a nonstationary predictive deconvolution code in Matlab based in equation (22). We call our method *slicedecon* and will use that name in the remainder of this paper. We will compare *slicedecon* to Gabor deconvolution and will use the term *gabordecon* for the latter. Features implemented in *slicedecon* include asymmetric POU's and the ability to prescribe the prediction filter parameters (operator length, lag,

and stability constant) arbitrarily with time. In any particular Gabor slice, the prediction filter is stationary as described by Wiener's theory; however, the deconvolved signal is constructed by the superposition of many such slices and is therefore nonstationary. At this time, it is not obvious how quickly the deconvolution parameters can vary in the final result nor precisely how the POU controls this. This is a subject of future investigation.

TESTING NONSTATIONARY PREDICTIVE DECONVOLUTION

Unit-lag testing

One of the basic features of ordinary predictive deconvolution is that it is theoretically equivalent to ordinary (Wiener spiking) deconvolution when the prediction distance is set to 1 sample. Thus, we expect *slicedecon* with unit prediction distance to be comparable to *gabordecon*. We do not expect this equivalence to be exact because *gabordecon* designs its nonstationary deconvolution operators simultaneously for all times, thus making them consistent with a physical attenuation model, while in *slicedecon* each operator is designed independently. Furthermore, *gabordecon* is essentially a frequency domain algorithm while *slicedecon* is entirely in the time domain.

As an initial comparison of these algorithms, we constructed a nonstationary synthetic seismogram from a random reflectivity using a forward Q filter (Q=50) and a minimum phase source signature (Figure 4). This is the typical synthetic that we have previously used to demonstrate the effectiveness of *gabordecon* for both source signature removal and inverse Q filtering. Figure 5 shows a comparison of both algorithms using the same symmetric POU and, for *slicedecon*, a variety of prediction operator lengths were tested. As a comparison, Figure 6 shows the performance of reasonably equivalent stationary algorithms on the same synthetic after automatic gain correction. For the stationary equivalent of *gabordecon* we chose frequency-domain spiking deconvolution (here called *fdecon*), while for *slicedecon* we used stationary unit-lag predictive deconvolution (called *prdecon*). To assess the success of any particular deconvolution, it should be compared to the reflectivity trace at the bottom of each figure. Several conclusions can be drawn from these figures: (1) the nonstationary algorithms perform substantially better than their stationary counterparts, (2) *gabordecon* achieves better temporal resolution than *slicedecon* in the time zone [0.2,0.8] sec, (3) *slicedecon* performs slightly better than *gabordecon* in the time zone [0.8,1.2] sec, and (4) both nonstationary and stationary predictive deconvolution are remarkably insensitive to prediction operator length.

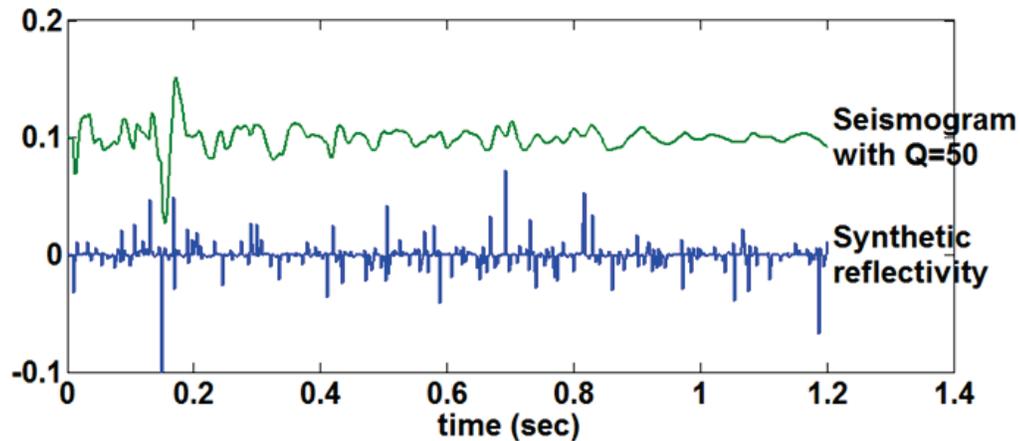


FIG. 4. A random reflectivity and the corresponding nonstationary synthetic seismogram are shown. The seismogram contains primary reflections only and was created by applying a forward Q filter ($Q=50$) to the reflectivity followed by a stationary convolution with a minimum phase source signature. A constant amplitude shift of 0.1 has been added to the seismogram to display it above the reflectivity.

To understand the performance of these algorithms in more detail, it is helpful to examine time-frequency decompositions of the salient results. The forward Gabor transform is the obvious tool and Figure 7 Gabor magnitude spectra for the input signal, the reflectivity, and *slicedecon*, *gabordecon*, *prdecon*, and *fdecon* results. By comparing the Gabor magnitude spectrum of the input signal with that of each deconvolution result, it is now apparent that both *slicedecon* and *gabordecon* have a considerable nonstationary action (meaning that they alter the input signal in both time and frequency), although the does considerably more nonstationary whitening than the former. Furthermore, the ideal deconvolution result should give the Gabor magnitude spectrum of the reflectivity and again it appears that *gabordecon* is superior. The reason for this superior performance is the subject of ongoing investigation but it must certainly be mostly because of differences in operator design. As mentioned previously, *gabordecon* designs a single, self-consistent nonstationary operator in the time-frequency domain, while *slicedecon* designs each operator independently in each window. It seems likely that the *slicedecon* algorithm can be extended to simultaneous and self-consistent design but this has not yet been done. The *prdecon* and *fdecon* results seem almost identical and this is consistent the essential equivalence of these methods (for unit-lag prediction). A similar statement of equivalence for the nonstationary algorithms is not presently attainable.

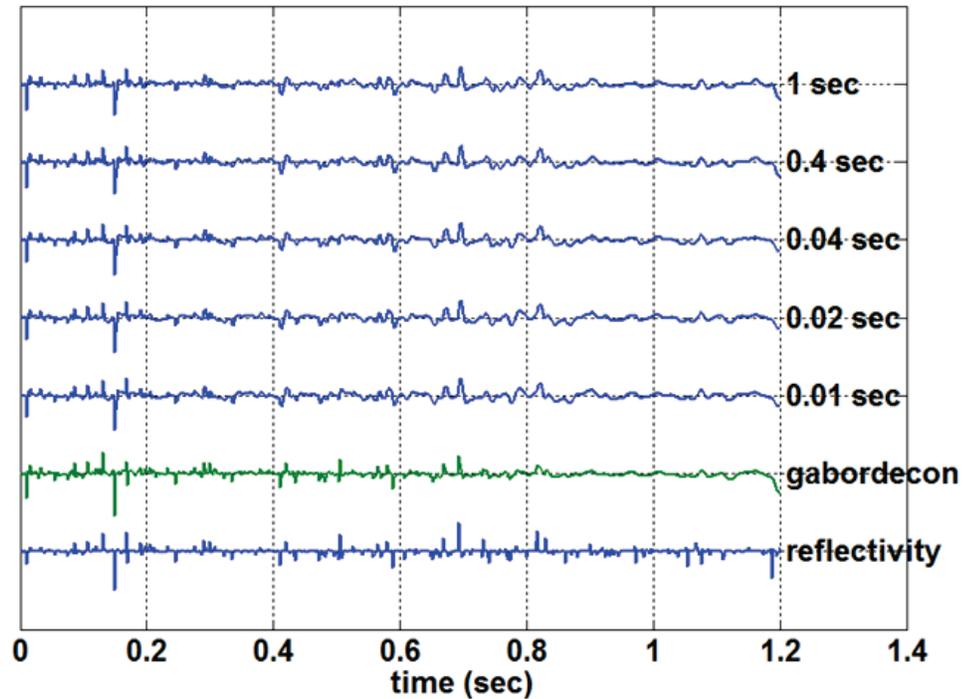


FIG. 5. A comparison between *gabordecon* and unit-lag *slicedecon*. The upper five traces are all *slicedecon* results where the numeric label on the right gives the prediction operator length. Both algorithms used symmetric Gaussians (as in Figure 1) with a 0.2 sec half-width at a spacing of 0.01 sec. The input to both algorithms was the synthetic seismogram of Figure 4 and the reflectivity shown is that of Figure 4.

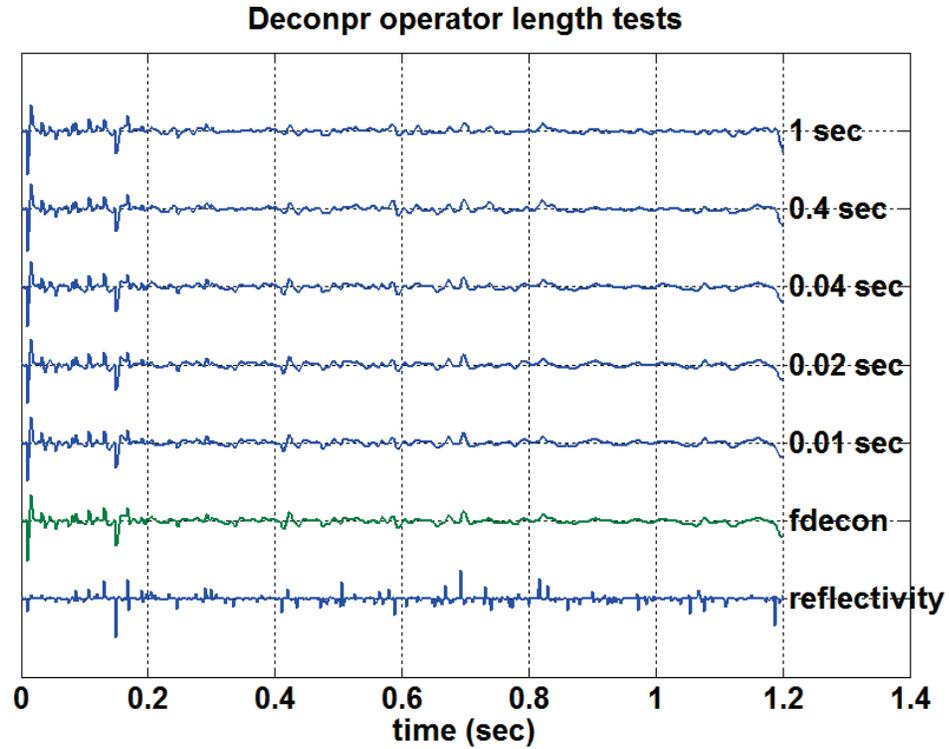


FIG. 6. Comparable to Figure 5 except that stationary frequency-domain deconvolution (*fdecon*) in place of *gabordecon* and stationary unit-lag predictive deconvolution (*prdecon*) is used in place of *slicedecon*. Prior to deconvolution the input seismogram (Figure 4) was gain corrected with a standard 0.2 sec AGC (automatic gain correction).

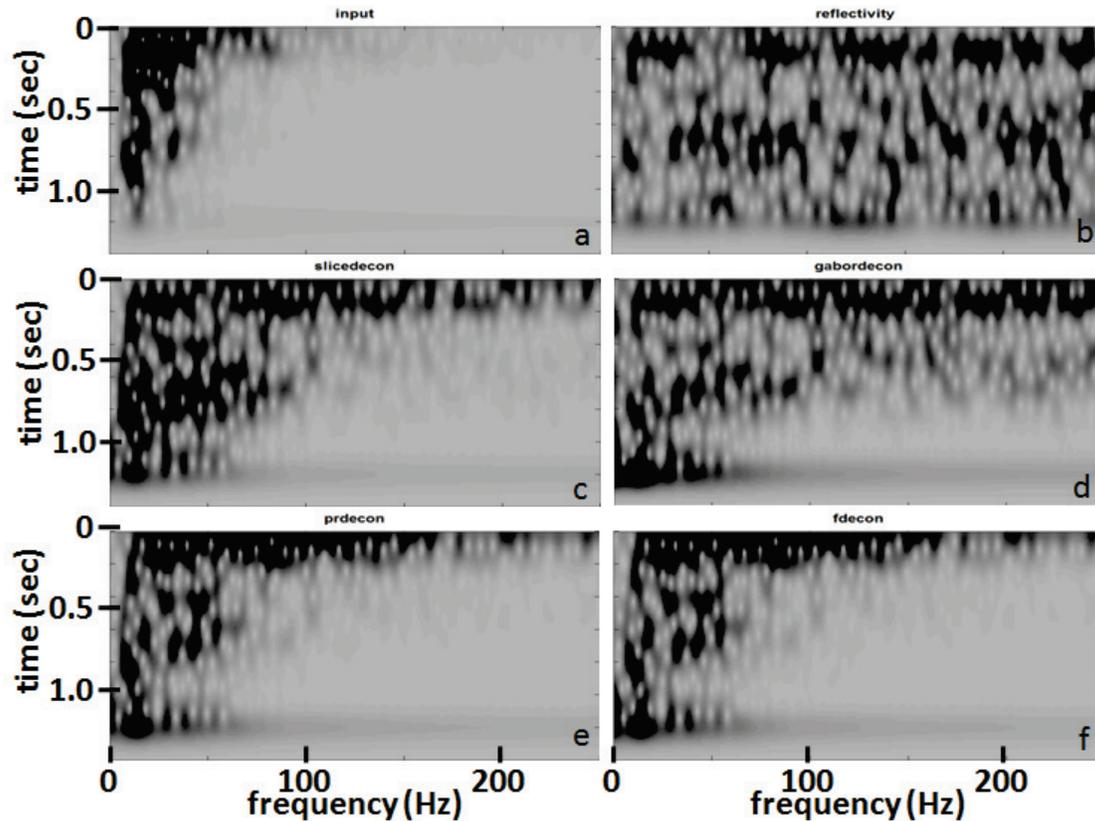


FIG. 7. Gabor magnitude spectra of: a) input to all of the deconvolution (see Figure 4), b) true reflectivity (again Figure 4), c) *slicedecon* result from Figure 5 (0.04 s operator length), d) *gabordecon* result of Figure 5, e) *prdecon* result of Figure 6 (0.04 s operator length), f) *fdecon* result of Figure 6.

Extension to general prediction lags

The original justification for prediction filtering at lags other than unity was that it was thought that multiples could be removed in this way. There has been some success in this regard; however, it was rapidly realized that multiples are a nonstationary effect for at least two reasons: (1) even in the case of a single reflector (e.g. the ocean bottom) the time lag between the primary reflection and its various multiples is only constant for zero source-receiver offset, (2) when a wave propagates through a finely layered medium, a multiple train accumulates behind the primary, and the length of this multiple train grows with propagation time. Therefore, the investigation of general prediction lags seems to require a synthetic trace (or gather) with considerable multiple content. For this reason, an elastic raytracing algorithm was employed to generate a synthetic gather for a stratified (i.e. variations in depth only) 3D model. Figure 8 shows the chosen model which includes an interval of alternating property values representing a coal-shale sequence. Figures 9 and 10 show the zero offset seismic response of the model with both primaries only (hereafter PO) and primaries-plus-multiples (hereafter PPM) traces shown. In this and all subsequent seismograms with multiples, all possible 3-bounce events are included. The elastic response represents the vertical component of displacement and there is no anelasticity (Q-effect) included. Examination of Figures 9 and 10 shows that the inclusion of multiples alters the seismic response considerably. The obvious effect is

that the length of the seismogram increases dramatically. Less obvious but still important are the subtle changes in the time zone of the primaries.

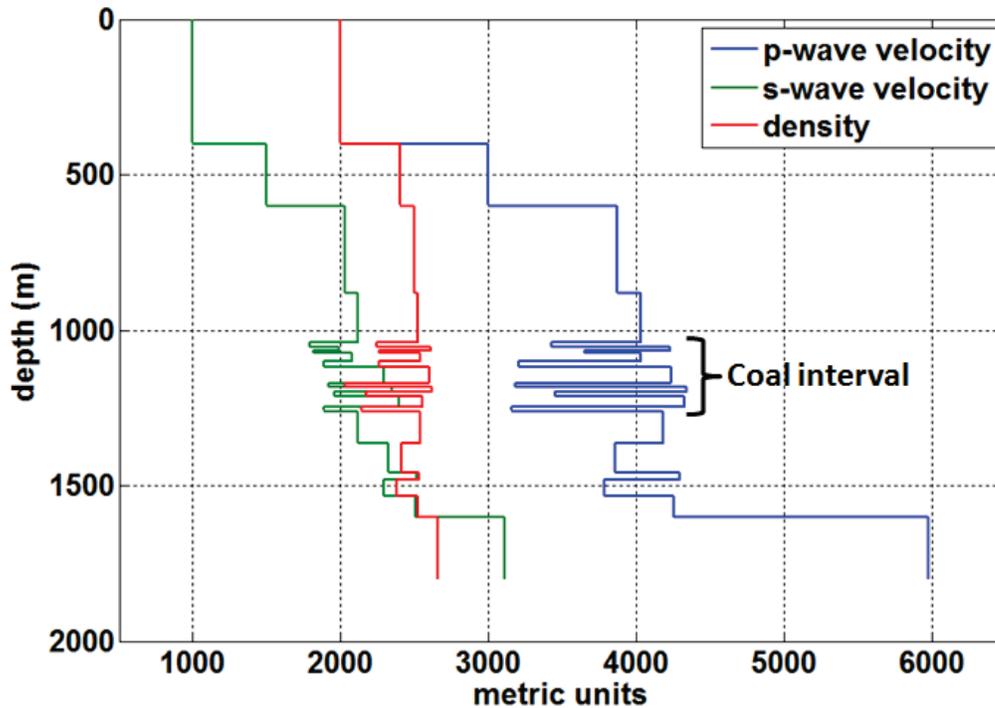


FIG. 8. The stratified (i.e. depth variable) 3D elastic model used to create a synthetic gather with considerable multiple content. Velocity values are in m/sec while density is in kg/m^3 .

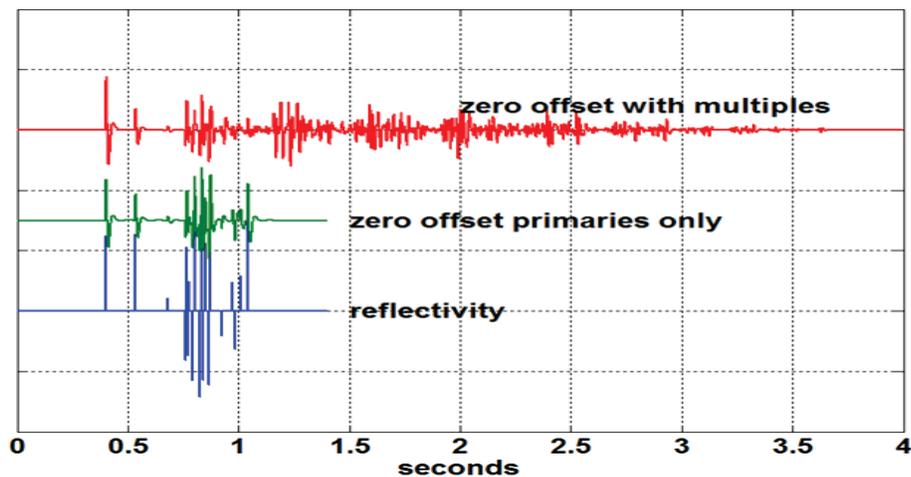


FIG. 9. The reflectivity (p-wave) and the zero-offset seismic responses (vertical component of displacement) for the model of Figure 8 are shown. The primaries only response is simply a wavelet convolved with reflectivity and therefore has no transmission losses or spherical divergence. The trace with multiples has these effects but has been gain corrected.

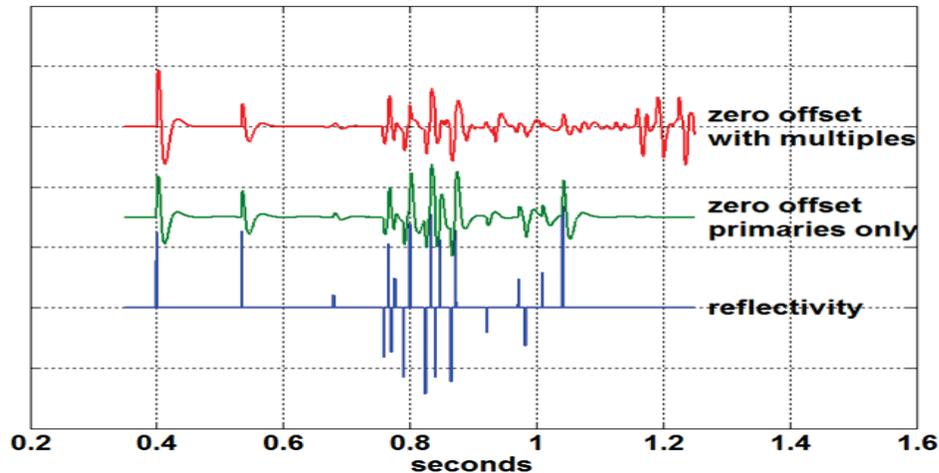


FIG. 10. An enlargement of the first portion of Figure 9. It is now possible to discern the minimum-phase source waveform and the details of the alteration of the seismogram due to multiple inclusion.

Figure 11 shows an initial deconvolution test of the PPM (primaries-plus-multiples) seismogram of Figure 10. This test compares *gabordecon* with *slicedecon* where the latter is run with a variety of prediction gaps. The 0.001 second prediction gap corresponds to unit-lag and the prediction operator length was 0.080 seconds. As noted previously, unit-lag *slicedecon* and *gabordecon* produce quite similar results but it is interesting that *slicedecon* did not invert the first wavelet. This is likely because the causal prediction operators must accumulate some trace samples before a successful prediction can be made. These results are in accord with those expected from experience with stationary predictive deconvolution. Visually, it appears that longer prediction gaps cause a reduction in the “level” of deconvolution. We also note that the spiking deconvolutions (*gabordecon* and *slicedecon* with 0.001s lag) are quite erroneous from about 0.8 s to 1.0 s. This is the zone where interbed multiples from the coal interval (Figure 8) can be expected to be a severe problem.

To help make this analysis more quantitative, we now examine a series of nonstationary autocorrelations (see equation (23)). Figure 12 shows the nonstationary autocorrelations (using a symmetric window with a 0.1 second half-width) of the two synthetic seismograms of Figure 10. Since the primaries-only seismogram is simply a convolution of a wavelet with a reflectivity, much of the structure in the nonstationary autocorrelation should be attributed to the reflectivity. This is because the wavelet is a stationary effect and should not give rise to a time-variant autocorrelation. Therefore, any nonstationary character is likely to be an effect of the reflectivity. This brings us into confrontation with a major limitation of the methods investigated here, and that is the reflectivity will almost never be statistically white. A so-called *white reflectivity* is defined to be one whose autocorrelation is a Dirac delta at the origin, or digitally, a spike at zero lag. Usually this is only approximately realized with finite-length reflectivity sequences, but the assumption is common in deconvolution methods. In a nonstationary method based on windowing, the localized reflectivity can be very non-white.

Essentially, if $r(t)$ is a reflectivity series with the property $a_r(t) = (r \cdot \bar{r})(t) \sim \delta(t)$, then the localized reflectivity has autocorrelation $a_{loc}(t) = (r_{loc} \cdot \bar{r}_{loc})(t)$ where $r_{loc} = gr$ with g being a suitable localizing window. Thus the window is inextricably linked with the definition of r_{loc} and hence strongly influences the autocorrelation. As an aid in understanding, Figure 13 repeats Figure 12 but with panel a) replaced by a primaries-only convolutional seismogram using a computer random reflectivity. Thus the two panels of Figure 13 use the same wavelet but the left panel uses a computer random reflectivity which the right panel uses the non-random reflectivity of Figure 9.

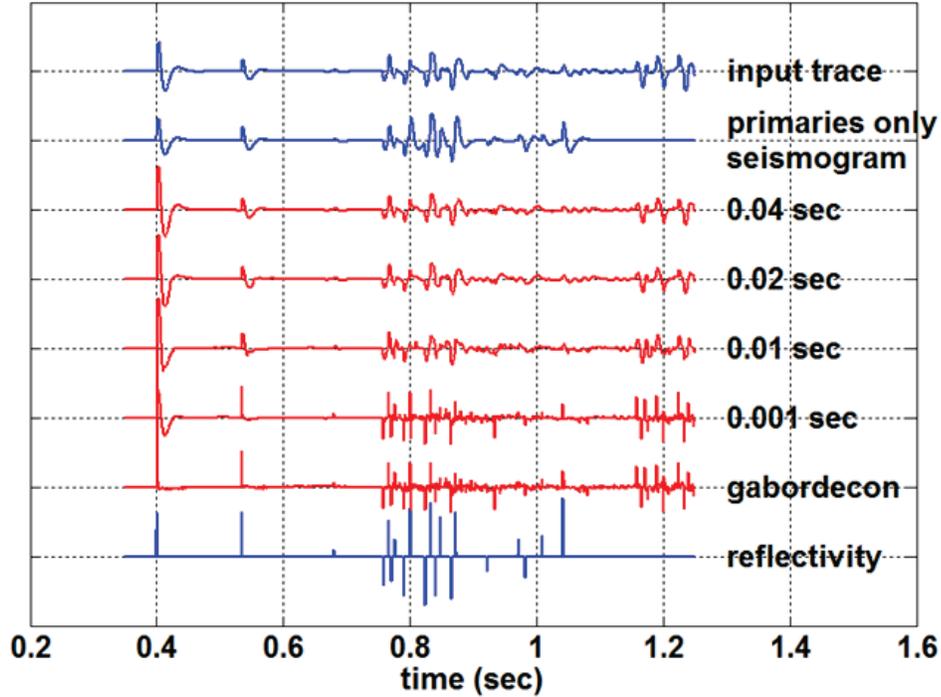


FIG. 11. A deconvolution test of the zero-offset seismogram with multiples of Figure 10. Red traces are deconvolution results and blue traces are from Figure 10 for comparison. Red traces with numeric labels are from *slicedecon* and the labels give the size of the prediction gap. The prediction operator length was 0.080 seconds.

Figure 14 shows nonstationary autocorrelations for the two spiking deconvolution results of Figure 11: *gabordecon* and *slicedecon* with 0.001 sec prediction lag. Since these should be reflectivity estimates, we should expect to see each row of these autocorrelations approximate a delta function, and this is reasonably true. The display makes this fact difficult to assess so Figure 15 plots the rows of Figure 14a superimposed on each other in a 1D amplitude display which makes it easy to see the spike at zero lag. We note that these autocorrelations are almost identical except for the times from about 0.35 to 0.55 seconds where the *slicedecon* result is underwhitened as discussed previously.

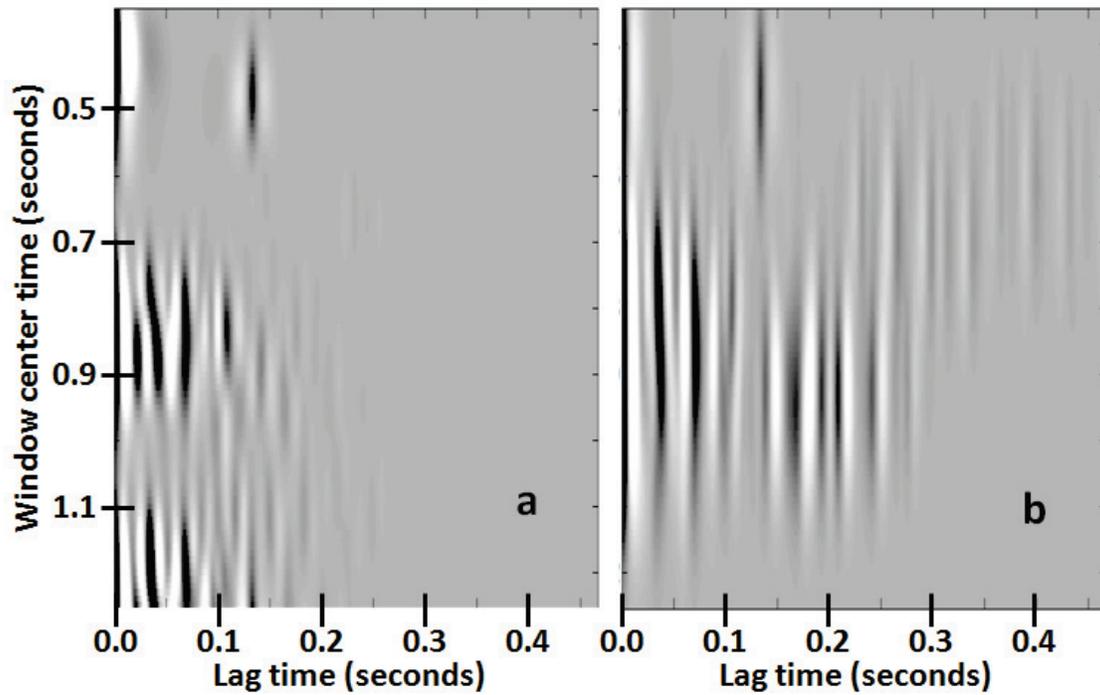


FIG. 12. Nonstationary autocorrelations are shown for a) The zero-offset with multiples trace of Figure 10, and b) The primaries only trace of Figure 10. The trace of panel a) was input to the deconvolution tests of Figure 11.

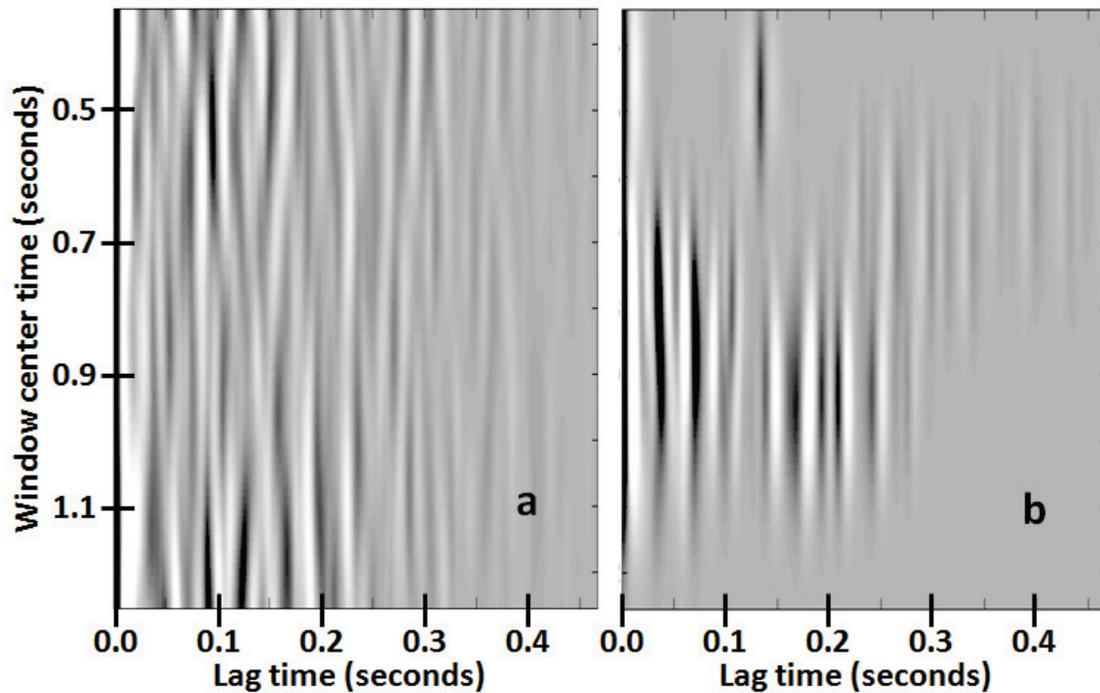


FIG. 13. Panel b) is identical to Figure 12b. Panel a) is the nonstationary autocorrelation from a computer random reflectivity convolved with the same wavelet as panel b).

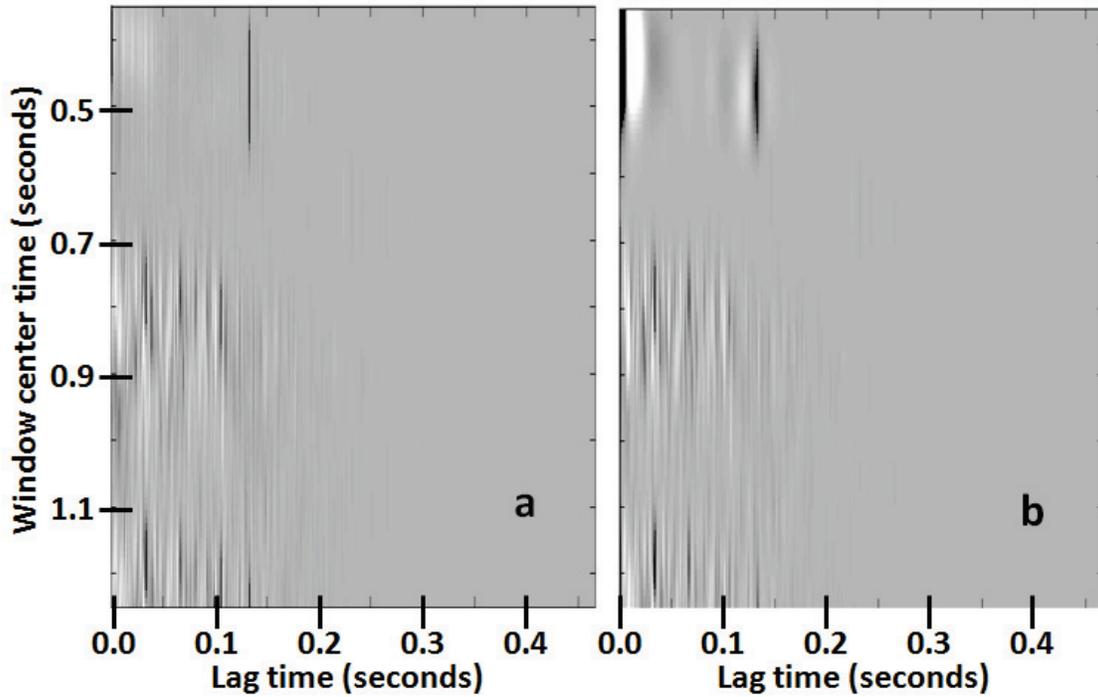


FIG. 14. Nonstationary autocorrelations are shown for a) The *gabordecon* result of Figure 11, and b) The *slicedecon* result with prediction lag of 0.001 seconds of Figure 11. It is difficult to discern that these panels have a very large, narrow peak at zero lag. See Figure 15 for interpretation assistance.

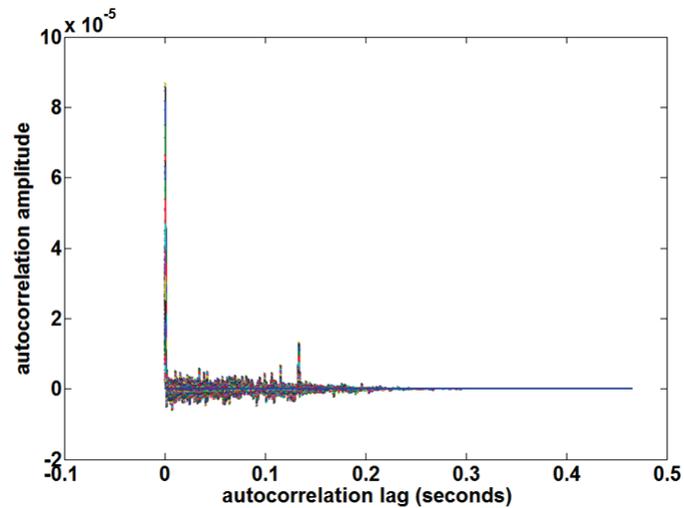


FIG. 15. The nonstationary autocorrelation of Figure 14 a) is shown plotted with all of its rows superimposed. Only the non-negative lags of the symmetric autocorrelation are shown. The autocorrelations are dominated by a large spike at zero lag.

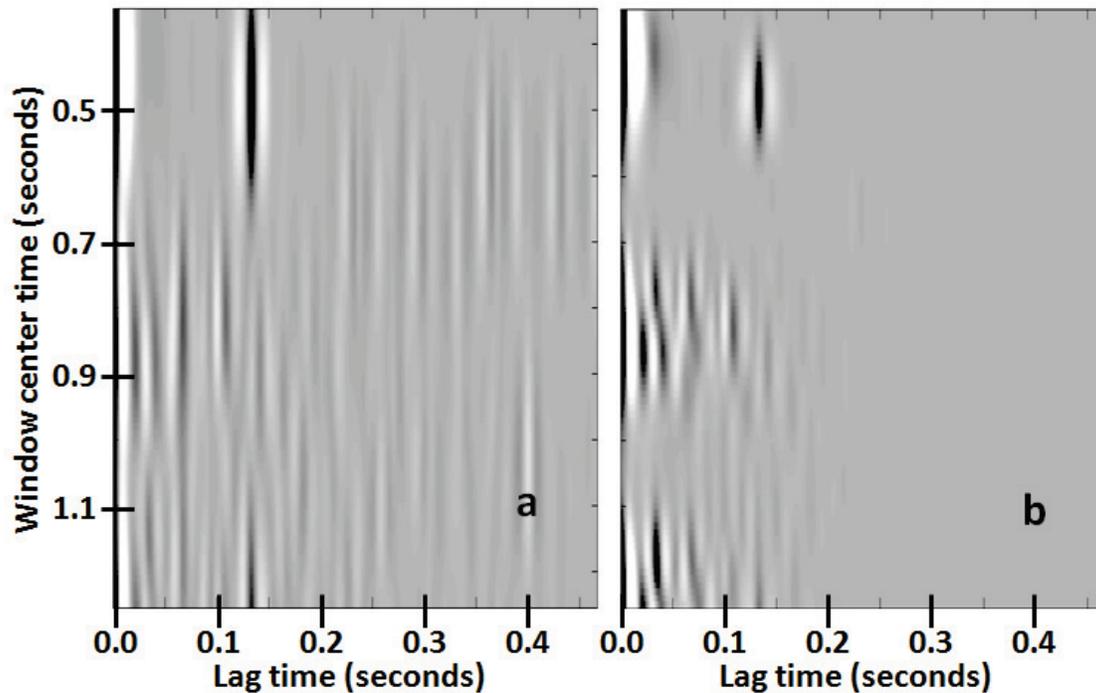


FIG. 16. Nonstationary autocorrelations are shown for a) the *slicedecon* result with a prediction gap of 0.020 seconds (Figure 11), and b) the *slicedecon* result with a prediction gap of 0.040 seconds (Figure 11). In both cases the prediction operator length was 0.080 seconds.

Figure 16 shows autocorrelations from two of the gapped *slicedecon* results of Figure 11. The expectation here is that gapped deconvolution with a prediction lag of t_{gap} and an operator length of t_{op} should suppress the autocorrelation for lags satisfying $t_{lag} \in [t_{gap}, t_{gap} + t_{op}]$. Comparison of these autocorrelations with that of Figure 12a shows that there does seem to be a reduction of strength in the appropriate lag windows ($[0.020, 0.100]$ for Figure 16a and $[0.040, 0.120]$ for Figure 16b).

Figure 17 shows an entire offset gather created from the model of Figure 8. This is the vertical component of displacement and includes all events of three reflections or fewer. The gather has had temporal gain applied. In Figures 18 and 19 we show *gabordecon* and *slicedecon* (in spiking mode). The results are comparable and entirely as expected based on the previous testing of a single trace. Both algorithms do a good job of whitening, although *gabordecon* seems stronger, and the sections differ somewhat in overall amplitude.

A common practice with stationary predictive deconvolution is to follow a gapped deconvolution with a spiking deconvolution. We show this sequence in Figures 20 and 21. Again there are few surprises given the previous testing. It does appear that the result in Figure 21 is somewhat different from that in Figure 19 although we cannot attribute the differences to multiple suppression.

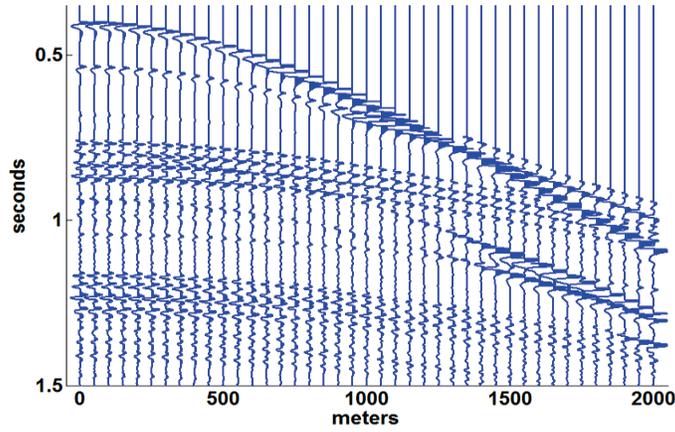


FIG. 17. A synthetic offset gather corresponding to the model of Figure 8. The vertical component of displacement is shown, the wavelet is 30 Hz. Dominant minimum phase, and all elastic events of three or fewer reflections are shown.

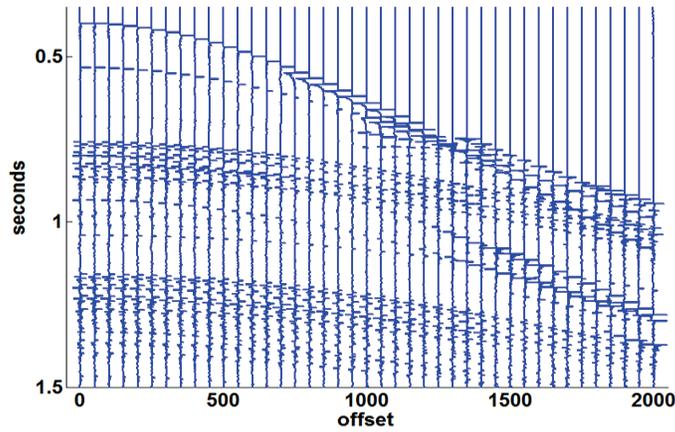


FIG. 18. The result of *gabordecon* on the data of Figure 17.

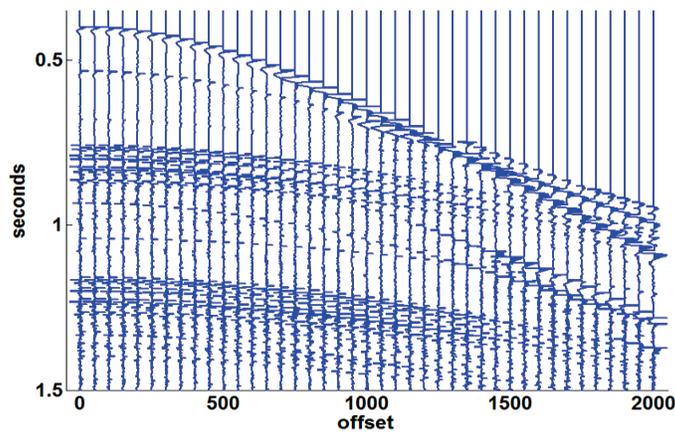


FIG. 19. The result of *slicedecon* run in spiking mode of the data of Figure 17.

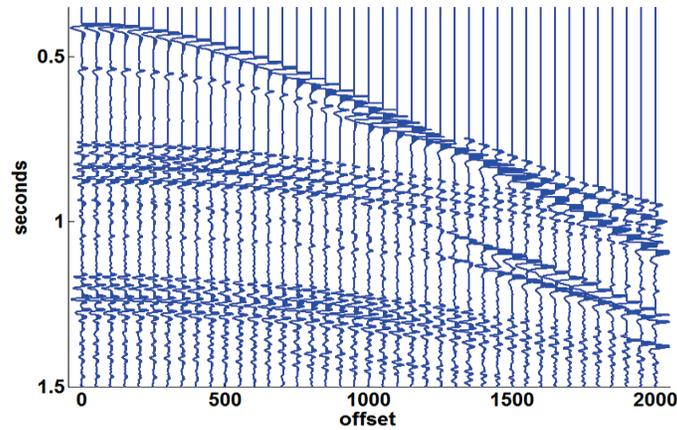


FIG. 20. The result of running *slicedecon* in gapped mode using a 0.1 second prediction gap and a 0.3 second operator length.

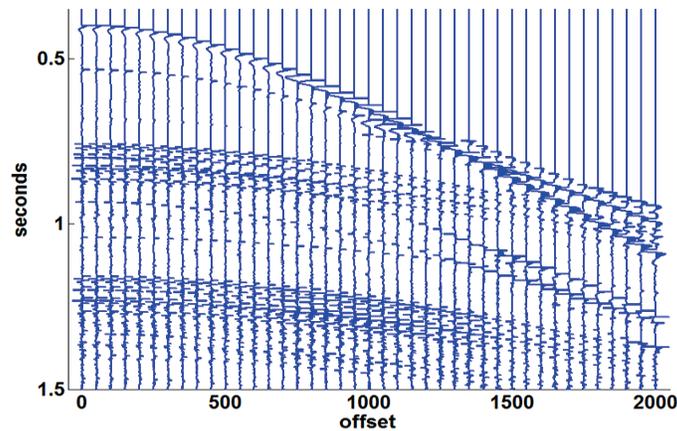


FIG. 21. The result of running *slicedecon* in spiking mode on the data of Figure 20. The prediction operator was 0.05 seconds long.

DISCUSSION

Stationary predictive deconvolution has almost fallen out of practice as other methods of multiple suppression have proven more effective. We conjectured that this was at least partially due to the nonstationary nature of the problem and hoped that a nonstationary implementation would help. Our initial attempts are reported here and, while not entirely successful, are still encouraging. We took the easy way here in that we simply implemented stationary prediction filtering in each Gabor window and made no attempt to link the operators in any physical way. Clearly, this is not the best that can be done. The operator in a later window should bear some relationship to that in earlier windows. In some sense the operators should progressively evolve in time. We have achieved something like this in *gabordecon* but there is no explicit inclusion of multiples there, only attenuation. Of course, short-path multiples lead to effects indistinguishable from Q and so *gabordecon* should be attacking those. However, multiples that lag behind the primary by more than the temporal length of the pulse will be treated as primaries. To progress beyond the current state, we need a better theory of nonstationary prediction filtering that incorporates more physics. This will be the subject of future research.

CONCLUSIONS

We have successfully developed a nonstationary predictive deconvolution by using stationary prediction filters on the Gabor slice decomposition of a signal. Called *slicedecon* our method compares reasonably well to *gabordecon* when the prediction distance is unity. That it is not quite as good as *gabordecon* is attributed to the fact that the deconvolution operators are designed independently rather than simultaneously. Evidence for the attenuations of multiples by *slicedecon* with a prediction distance greater than one sample is presently slim. We observe some reduction of a given periodicity in a nonstationary autocorrelation but can see little improvement in the final deconvolution. Still, we remain optimistic that further work will improve the method.

ACKNOWLEDGEMENTS

We thank the Sponsors of CREWES for their continued support and also thank NSERC, MITACS, and the sponsors of POTSI. We thank Pat Daley for providing the multiple-contaminated synthetic seismogram used in this study.

REFERENCES

- Margrave, G., and Lamoureux, M., 2001, Gabor deconvolution: The CREWES Project Research Report, 13, 241-276.
- Margrave, G. F., Gibson, P. C., Grossman, J. P., Henley, D. C., Iliescu, V., and Lamoureux, M. P., 2004, The Gabor Transform, pseudodifferential operators, and seismic deconvolution, *Integrated Computer-Aided Engineering*, **9**, 1-13.
- Peacock, K. L., and Treitel, S., 1969, Predictive deconvolution: Theory and practice: *Geophysics*, **63**, 155-169.
- Perez, M. A., and Henley, D. C., 2000, Multiple attenuation via predictive deconvolution in the radial domain: in the 12th Annual Research Report of the CREWES Project.
- E. A. Robinson, and S. Treitel, 1967, Principles of digital Wiener filtering, *Geophys. Prosp.*, **15**, 311-333.
- Taner, M.T., 1980, Long-period sea-floor multiples and their suppression: *Geophys. Prosp.*, **28**, 30-48.